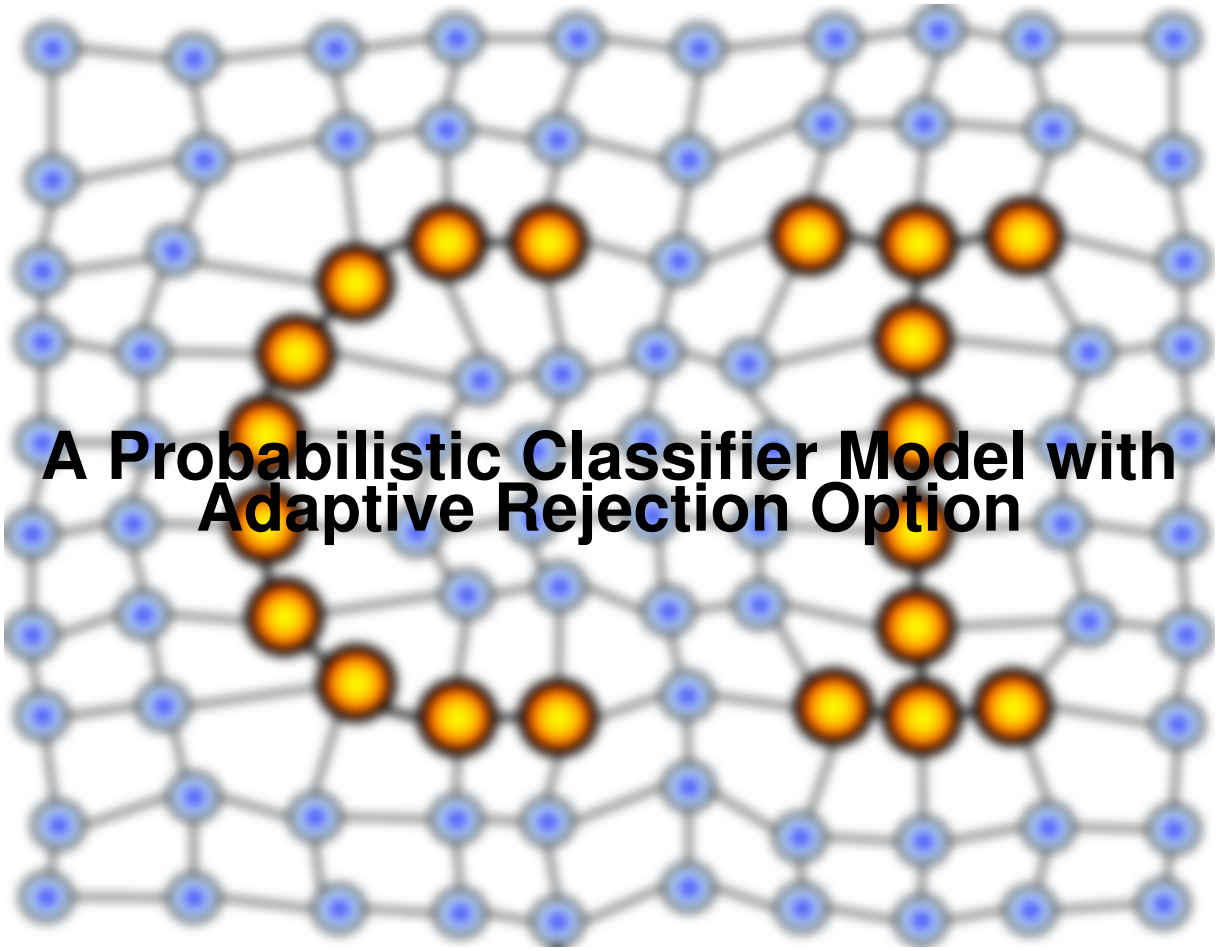
MACHINE LEARNING REPORTS



A Probabilistic Classifier Model with Adaptive Rejection Option

Report 01/2016
Submitted: 19.02.2016
Published: 20.02.2016

Lydia Fischer[1,2], and Thomas Villmann[3]
(1) - Bielefeld University, Universitätsstraße 25, 336615 Bielefeld, Germany
(2) - Honda Research Institute Europe, Carl-Legien-Straße 30, 63065 Offenbach, Germany
(3)- University of Applied Sciences Mittweida, 09648 Mittweida, Germany

Abstract

Lately the topic of rejecting decisions in a classification scenario became attention, e. g. in medical data analysis, since not only the decision itself but also the certainty of the decision is important. While often a reject option is used on top of a trained model, recent approaches include it directly in the objective function of the desired model, e. g. for learning vector quantization. Following this trend, we propose a theoretical framework for probabilistic models, e. g. Gaussian mixture models, which includes costs for wrong classification as well as costs for rejection in its objective function. Further the rejection threshold is optimised during the training phase of the model. The proposed method follows the ideas of C. K. Chows paper: *On optimum recognition error and reject tradeoff (1970)*. This article describes the new model in detail including the derivatives of the objective function. Keywords: classification, probabilistic model, rejection, adaptive threshold

A Probabilistic Classifier Model with Adaptive Rejection Option

Lydia Fischer[1,2], and Thomas Villmann[3] *

1 - Bielefeld University, Universitätsstraße 25, 336615 Bielefeld, Germany

2 - Honda Research Institute Europe, Carl-Legien-Straße 30, 63065 Offenbach, Germany

3 - University of Applied Sciences Mittweida, 09648 Mittweida, Germany

Abstract – **Lately the topic of rejecting decisions in a classification scenario became attention, e. g. in medical data analysis, since not only the decision itself but also the certainty of the decision is important. While often a reject option is used on top of a trained model, recent approaches include it directly in the objective function of the desired model, e. g. for learning vector quantization. Following this trend, we propose a theoretical framework for probabilistic models, e. g. Gaussian mixture models, which includes costs for wrong classification as well as costs for rejection in its objective function. Further the rejection threshold is optimised during the training phase of the model. The proposed method follows the ideas of C. K. Chows paper: *On optimum recognition error and reject tradeoff* (1970). This article describes the new model in detail including the derivatives of the objective function.**

Keywords: classification, probabilistic model, rejection, adaptive threshold

*L. Fischer acknowledges funding by the CoR-Lab Research Institute for Cognition and Robotics and gratefully acknowledges the financial support from the Honda Research Institute Europe.

1 Introduction

Secure classification is one of the key issues for many classification systems, for example in medical assistance diagnostic systems or forensic analytic systems. In this context, also cost-based classification decision systems nowadays get more and more attention. For those systems misclassification may generate expansive costs. However, also correct classification may be lead to costs, for example for medical treatment, which usually are much cheaper than the costs of misclassification. To avoid misclassification a good idea is to reject those samples, for which the classification decision shows a valuable uncertainty and feed them to a further investigation. The respective costs are usually smaller than misclassification costs.

Frequently, for a given classification task, a classifier model is selected or/and trained first and then applied in practice [1, 2]. Un-secure samples may be rejected in the application phase due to certain criteria. For example, if the data to be classified are assumed to be described as real valued feature vectors and the classification decision is based on distance evaluations in the data space, a data sample could be rejected if it is detected to be localized between two data classes [3, 4]. Yet, this approach has at least two drawbacks. First, the reject decision has to be adjusted to deal with the structure of the data space in that way that it reflects the assumed uncertainty accordingly by respective data distance decision in the feature space. Second, the model selection or/and the model training does not take the reject option into account to adjust the classifier model accordingly. Thus, the model application task differs from the training objective.

To overcome the latter problem, recently adaptive classifiers were proposed, which explicitly include reject options during model training [5, 6, 7, 8]. However, these models base their adaptation on geometric decisions, i. e. distance-based reject options are adapted to avoid misclassifications. In contrast, Chow proposed a cost-based classifier based on the *expected* costs regarding the misclassification, reject and correct classification costs [9, 10]. This model can be combined with the learning vector quantization (LVQ) approach yielding a LVQ with self-adjusting reject option (Reject-LVQ), which provides a robust prototype-based classifier with good performance [11, 12]. The disadvantage of this LVQ-approach is that it is not longer a probabilistic model, because the class distributions are in LVQ

approximately represented by prototype vectors distributed in the data space [13]. Yet, there exists a probabilistic variant of LVQ - the robust soft LVQ (RSLVQ,[14]). Thus, the aim of this paper is to adopt the ideas for prototype-based RSLVQ for the Chow-model in case of cost-based reject options. Incorporating further the idea of self-adjusting reject thresholds from Reject-LVQ, we obtain a probabilistic classifier model with self-adjusting reject threshold based on costs.

2 Model description

We start with a description of the Chow-approach for a (binary) classifier including a reject option. It provides an optimal model in the sense of a Bayesian decision [10] based on expected costs. For this purpose, we assume costs C_e for a misclassified data point, C_c the cost for a correctly classified sample as well as costs C_r for a rejected data point. Without loss of generality we can take $C_c = 0$, which can be always achieved by rescaling of C_e and C_r accordingly. Further, we do not consider asymmetric classification costs as discussed in [15]. However, a respective extension of the approach is straight forward.

Suppose a data set X with elements $(\mathbf{x}, y) \in \mathbb{R}^M \times \{1, \ldots, Z\}$ and a data point $\mathbf{x}$ belongs to one of the Z classes. We use a class-wise Gaussian mixture model consisting of ξ Gaussians

$$p(\mathbf{x}|\mathbf{w}_j) = K_j \cdot \exp(f_\sigma(\mathbf{x}, \mathbf{w}_j)); \quad f_\sigma(\mathbf{x}, \mathbf{w}_j) = -\frac{\|\mathbf{x} - \mathbf{w}_j\|}{2\sigma^2}$$

located at $\mathbf{w}_j \in \mathbb{R}^M$ to model the data. These play the role of prototypes as known from RSLVQ and are equipped with class labels $c_j \in \{1, \ldots, Z\}$. The parameter K_j is interpreted as the prior of the Gaussian and the parameter $\sigma > 0$ denotes its constant variance being isotropically for all dimensions. This latter assumption was made for simplicity. Generalizations towards a covariance matrix Σ instead of a constant σ are possible.

Using this Gaussian model, the probability of observing a data sample $\mathbf{x}$ is given by

$$p(\mathbf{x}) = \sum_{z=1}^{Z} p_z \cdot p_z(\mathbf{x})$$

whereas p_z is the prior of class z and

$$p_z(\mathbf{x}) = \sum_{j=1}^{\xi} \delta_z^{c_j} \cdot p(\mathbf{w}_j) \cdot p(\mathbf{x}|\mathbf{w}_j)$$

is the conditional probability that the class z has generated the data point $\mathbf{x}$. Here, the Kronecker-symbol $\delta_z^{c_j}$ is used with $\delta_z^{c_j} = 1$ if $z = c_j$ and 0 otherwise. The parameter $p(\mathbf{w}_j)$ is the prior of the corresponding Gaussian.

According to Chow [10], we apply a rejection in dependence on the quantity

$$m(\mathbf{x}, \mathbf{w}) = \frac{\max\limits_{z=1,\ldots,Z} \{p_z \cdot p_z(\mathbf{x})\}}{p(\mathbf{x})} \; . \tag{1}$$

In fact, a data point $\mathbf{x}$ has to be rejected if $m(\mathbf{x}, \mathbf{w}) < 1 - \theta$, i. e. $\theta \in (0, 1)$ denotes a rejection threshold. We introduce

$$\tilde{m}(\mathbf{x}, \mathbf{w}, \theta) = m(\mathbf{x}, \mathbf{w}) - 1 + \theta \tag{2}$$

with $\tilde{m}(\mathbf{x}, \mathbf{w}, \theta) \geq 0$ holds for accepted data points and $-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta) > 0$ is valid for rejected data. Thus, the Heaviside function $H(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))$ is used to count the accepted points while $H(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))$ detects the rejected points.

Using these quantities we can calculate the probability for an incorrect classification as

$$E(\mathbf{x}, y, \mathbf{w}) = \sum_{\substack{z=1 \\ z \neq y}}^{Z} p_z \cdot p_z(\mathbf{x}), \;\; \text{if } \tilde{m}(\mathbf{x}, \mathbf{w}, \theta) \geq 0 \tag{3}$$

whereas the probability of a rejected data point reads as

$$R(\mathbf{x}, \mathbf{w}) = \sum_{z=1}^{Z} p_z \cdot p_z(\mathbf{x}), \;\; \text{if } \tilde{m}(\mathbf{x}, \mathbf{w}, \theta) < 0 \; .$$

Collecting the previous observations, we can write the cost function for the classification model according to Chow as

$$E_{\text{cost}} = \sum_{(\mathbf{x}, y) \in X} (H(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta)) \cdot E(\mathbf{x}, y, \mathbf{w}) \cdot C_e + H(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta)) \cdot R(\mathbf{x}, \mathbf{w}) \cdot C_r) \rightarrow \min$$

incorporating the costs for misclassification C_e and C_r for a rejected data sample [10]. It balances the tradeoff between expected misclassification and reject costs

in dependence on the prototype distribution as well as the reject threshold θ. Hence, minimization of the cost function is the desired goal. This task can be performed using a stochastic gradient descend (SGD) thereon with respect to both the prototypes $\mathbf{w}_j$ and the rejection threshold θ according to the derivatives

$$\Delta \mathbf{w}_k \propto \frac{\partial E_{\text{cost}}}{\partial \mathbf{w}_k} \qquad\qquad \Delta\theta \propto \frac{\partial E_{\text{cost}}}{\partial \theta}$$

and keeping in mind that the determined quantities of the cost function E_{cost} depend explicitly on the prototypes $\mathbf{w}_j$ and the rejection threshold θ. However, to calculate these derivatives, we have to replace the Heaviside and the maximum function by suitable soft approximations. Doing so, both derivatives can be calculated. The respective formulae are finally derived in Appendix B2 as (11) and (12).

3 Conclusion

In this technical note we consider a SGD scheme for a Bayes classifier with reject option according to the Chow-model. It minimizes the expected cost in dependence on the misclassification and rejection costs. For this purpose, the class distributions are estimated by Gaussian mixtures. The centers of the Gaussians are adapted by gradient learning as well as the respective reject threshold. The explicit SGD learning rules are derived using suitable approximations for the Heaviside and maximum function, both contained in the original cost function.

In the next step, we will perform numerical simulations to study the numerical stability of the derived model. This could also include neighborhood cooperativeness between the Gaussians for regularization as it is known from the neural gas algorithm [20].

Furthermore, the replacement of the Gaussian mixture model for the approximation of the class distribution will be considered, for example, Student-t distributions are of interest following the inspiration by t-stochastic neighbor embedding [16] or general similarity measures obeying the probability properties [17].

Acknowledgment

The authors thank MARIKA KADEN und DAVID NEBEL, both from the Computational intelligence Group of the University of Applied Sciences Mittweida, for stimulating discussions and inspiring drinks.

Appendix

In this appendix we explain how the formal derivatives $\frac{\partial E_{\text{cost}}}{\partial \mathbf{w}_k}$ and $\frac{\partial E_{\text{cost}}}{\partial \theta}$ can be calculated. For this purpose, first some needed approximations are introduced. Thereafter, we carry out the derivatives applying the product rule and the chain rule for derivations. This is done step by step for all quantities contributing to the cost function E_{cost}. The final results of these calculation are given in Appendix B2 by the formulae (11) and (12) as the respective update rules.

Appendix A - Approximations

Since we want the derivatives of E_{cost} we need differentiable approximations for both the Heaviside function $H(\cdot)$ and the maximum function in (1).

A common smooth approximation of the Heaviside function is the sigmoid function

$$H(x) \approx \text{sig}_\zeta(x) = \frac{1}{1+\exp(\frac{x}{\zeta})} \tag{4}$$

with its derivative

$$\frac{\partial \text{sig}_\zeta(x)}{\partial x} = \frac{1}{\zeta} \cdot \text{sig}_\zeta(x) \cdot \left[1 - \text{sig}_\zeta(x)\right] .$$

The parameter $\zeta > 0$ determines the slope. In the limit $\zeta \searrow 0$ the sigmoid $\text{sig}_\zeta(x)$ converges to the Heaviside function $H(x)$.

An approximation of the maximum function is given with the α-softmax function discussed in [18]:

$$\max_{z=1,\ldots,Z} \{x_z\} \approx \alpha\text{-softmax}_{z=1,\ldots,Z} \{x_z\} = S_\alpha [\![x_z]\!] = \frac{\sum_{z=1}^{Z} x_z \cdot \exp(\alpha \cdot x_z)}{\sum_{z=1}^{Z} \exp(\alpha \cdot x_z)} \tag{5}$$

with $\alpha > 0$ as an approximation parameter and $[\![x_z]\!] = \{x_1, \ldots, x_Z\}$. The greater the α-value the better is the approximation of the maximum function. The derivative of $S_\alpha(\mathbf{x})$ is obtained as

$$\frac{\partial S_\alpha [\![x_z]\!]}{\partial x_k} = \frac{\exp(\alpha \cdot x_k)}{\sum_{z=1}^{Z} \exp(\alpha \cdot x_z)} \left[1 - \alpha \left(x_k - S_\alpha [\![x_z]\!]\right)\right] \tag{6}$$

as demonstrated in [19, 18].

The explained approximations (4) and (5) allow to calculate the required derivations of the cost function.

Appendix B1 - Derivatives of the cost function with respect to θ and $\mathbf{w}_k$ (first part)

When considering the derivatives for the cost function, we are looking for

$$\frac{\partial E_{\text{cost}}}{\partial \theta} = \sum_{(\mathbf{x},y)\in X} \left[C_e \cdot E(\mathbf{x},y,\mathbf{w}) \cdot \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \theta} + C_r \cdot R(\mathbf{x},\mathbf{w}) \cdot \frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \theta} \right]$$

$$\begin{aligned}\frac{\partial E_{\text{cost}}}{\partial \mathbf{w}_k} = \sum_{(\mathbf{x},y)\in X} \Bigg(& C_e \left[\frac{\partial H(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \cdot E(\mathbf{x},y,\mathbf{w}) + H(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot \frac{\partial E(\mathbf{x},y,\mathbf{w})}{\partial \mathbf{w}_k} \right] \\ & + C_r \left[\frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \cdot R(\mathbf{x},\mathbf{w}) + H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot \frac{\partial R(\mathbf{x},\mathbf{w})}{\partial \mathbf{w}_k} \right] \Bigg) \ .\end{aligned}$$

The following paragraphs contain the derivations of the single parts needed for the derivatives of the cost function. The final result is collected in Appendix B2, formulae (11) and (12).

Derivatives of $E(\mathbf{x},y,\mathbf{w})$, $R(\mathbf{x},\mathbf{w})$ with respect to $\mathbf{w}_k$

Since the contribution of a prototype with the same label as the original class y of a data point $\mathbf{x}$ is excluded in the sum of $E(\mathbf{x},y,\mathbf{w})$ (3), the derivative of $E(\mathbf{x},y,\mathbf{w})$ with respect to $\mathbf{w}_k$ has two parts:

$$\frac{\partial E(\mathbf{x},y,\mathbf{w})}{\partial \mathbf{w}_k} = \begin{cases} p_{c_k} \cdot \frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k}, & c_k \neq y \\ 0, & \text{else} \end{cases} \tag{7}$$

and for $R(\mathbf{x},\mathbf{w})$ we get:

$$\frac{\partial R(\mathbf{x},\mathbf{w})}{\partial \mathbf{w}_k} = p_{c_k} \cdot \frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k} \ .$$

In both equations the term $\frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k}$ has to be clarified.

Derivatives of $p(\mathbf{x})$, $p_z(\mathbf{x})$ with respect to $\mathbf{w}_k$:

$$\begin{aligned}\frac{\partial p_z(\mathbf{x})}{\partial \mathbf{w}_k} &= \delta_z^{c_k} \cdot p(\mathbf{w}_k) \cdot \frac{\partial p(\mathbf{x}|\mathbf{w}_k)}{\partial \mathbf{w}_k} \\ &= \delta_z^{c_k} \cdot p(\mathbf{w}_k) \cdot K_k \cdot \exp(f_\sigma(\mathbf{x}, \mathbf{w}_k)) \cdot \frac{\partial f_\sigma(\mathbf{x}, \mathbf{w}_k)}{\partial \mathbf{w}_k} \\ &= \delta_z^{c_k} \cdot p(\mathbf{w}_k) \cdot K_k \cdot \exp(-\frac{\|\mathbf{x}-\mathbf{w}_k\|}{2\sigma^2}) \cdot (-\frac{1}{\sigma^2}) \cdot (\mathbf{x}-\mathbf{w}_k)\end{aligned}$$

$$\frac{\partial p(\mathbf{x})}{\partial \mathbf{w}_k} = p_{c_k} \cdot \frac{\partial p_z(\mathbf{x})}{\partial \mathbf{w}_k}$$

Derivatives of the Approximation of the Heaviside function with respect to θ and $\mathbf{w}_k$:

Note that we use $H(\pm\tilde{m}(\mathbf{x}, \mathbf{w}, \theta)) \approx \text{sig}_\zeta(\pm\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))$. Then we derive the following derivatives with respect to θ:

$$\begin{aligned}\frac{\partial H(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial \theta} \approx & \frac{\partial \text{sig}_\zeta(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial \theta} \\ = & \frac{\partial \text{sig}_\zeta(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial \tilde{m}(\mathbf{x}, \mathbf{w}, \theta)} \cdot \frac{\partial \tilde{m}(\mathbf{x}, \mathbf{w}, \theta)}{\partial \theta} \\ = & \frac{\partial \text{sig}_\zeta(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial \tilde{m}(\mathbf{x}, \mathbf{w}, \theta)} \cdot 1 \\ = & \frac{1}{\zeta} \cdot \text{sig}_\zeta(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta)) \cdot \Big[1 - \text{sig}_\zeta(\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))\Big]\end{aligned}$$

and

$$\begin{aligned}\frac{\partial H(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial \theta} \approx & \frac{\partial \text{sig}_\zeta(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial \theta} \\ = & \frac{\partial \text{sig}_\zeta(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial (-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))} \cdot \frac{\partial (-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial \theta} \\ = & \frac{\partial \text{sig}_\zeta(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))}{\partial (-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))} \cdot (-1) \\ = & -\frac{1}{\zeta} \cdot \text{sig}_\zeta(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta)) \cdot \Big[1 - \text{sig}_\zeta(-\tilde{m}(\mathbf{x}, \mathbf{w}, \theta))\Big] \ .\end{aligned}$$

Further we can obtain the following derivatives of the Heaviside function with respect to $\mathbf{w}_k$:

$$\begin{aligned}\frac{\partial H(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \approx & \frac{\partial \mathrm{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \\ = & \frac{\partial \mathrm{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)} \cdot \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \\ = & \frac{1}{\zeta} \cdot \mathrm{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot \Big[1 - \mathrm{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta))\Big] \cdot \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k}\end{aligned}$$

and

$$\begin{aligned}\frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \approx & \frac{\partial \mathrm{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \\ = & \frac{1}{\zeta} \cdot \frac{\partial \mathrm{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))} \cdot \frac{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \\ = & \mathrm{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \Big[1 - \mathrm{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))\Big] \frac{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} \,.\end{aligned}$$

The terms $\frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k}$ and $\frac{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k}$ are calculated in the next step.

Derivatives of $\tilde{m}(\mathbf{x},\mathbf{w},\theta)$ with respect to θ and $\mathbf{w}_k$: Remember, we introduced the quantity $\tilde{m}(\mathbf{x},\mathbf{w},\theta) = m(\mathbf{x},\mathbf{w}) - 1 + \theta$ in (2). Hence, we get for their derivatives with respect to θ

$$\frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \theta} = 1 \;\text{ and }\; \frac{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \theta} = -1\,.$$

For the calculation of the derivative of $\tilde{m}(\mathbf{x},\mathbf{w},\theta)$ with respect to $\mathbf{w}_k$ we first observe that

$$\frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} = \frac{\partial m(\mathbf{x},\mathbf{w})}{\partial \mathbf{w}_k}$$

is valid. Now we approximate in (1) the maximum function by the α-softmax function (5) such that

$$m(\mathbf{x},\mathbf{w}) \approx \frac{S_\alpha\,[\![p_z \cdot p_z(\mathbf{x})]\!]}{p(\mathbf{x})}$$

where

$$[\![p_z \cdot p_z(\mathbf{x})]\!] = \{p_z \cdot p_z(\mathbf{x})\}_{z=1,\ldots Z}\,.$$

In this way we obtain

$$
\begin{aligned}
&\frac{\partial m(\mathbf{x},\mathbf{w})}{\partial \mathbf{w}_k} \\
\approx &\frac{\partial}{\partial \mathbf{w}_k}\left[\frac{S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right]}{p(\mathbf{x})}\right] \\
= &\frac{\frac{\partial S_\alpha(p_z \cdot p_z(\mathbf{x}))}{\partial \mathbf{w}_k} \cdot p(\mathbf{x}) - S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right] \cdot \frac{\partial p(\mathbf{x})}{\partial \mathbf{w}_k}}{(p(\mathbf{x}))^2} \\
= &\frac{\frac{\partial S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right]}{\partial (p_z \cdot p_z(\mathbf{x}))} \cdot \frac{\partial (p_z \cdot p_z(\mathbf{x}))}{\partial \mathbf{w}_k} \cdot p(\mathbf{x}) - S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right] \cdot p_{c_k} \cdot \frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k}}{(p(\mathbf{x}))^2} \\
= &\frac{\frac{\partial S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right]}{\partial (p_z \cdot p_z(\mathbf{x}))} \cdot p_{c_k} \cdot \frac{\partial (p_{c_k}(\mathbf{x}))}{\partial \mathbf{w}_k} \cdot p(\mathbf{x}) - S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right] \cdot p_{c_k} \cdot \frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k}}{(p(\mathbf{x}))^2} \\
= &\frac{p_{c_k}}{(p(\mathbf{x}))^2} \cdot \frac{\partial (p_{c_k}(\mathbf{x}))}{\partial \mathbf{w}_k} \cdot \left(\frac{\partial S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right]}{\partial (p_z \cdot p_z(\mathbf{x}))} \cdot p(\mathbf{x}) - S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right]\right) \\
\overset{(6)}{=} &\frac{p_{c_k}}{(p(\mathbf{x}))^2} \cdot \frac{\partial (p_{c_k}(\mathbf{x}))}{\partial \mathbf{w}_k} \cdot \\
&\left(\frac{\exp(\alpha \cdot p_{c_k} \cdot p_{c_k}(\mathbf{x}))}{\sum_{z=1}^{Z} \exp(\alpha \cdot p_z \cdot p_z(\mathbf{x}))}\left[1 - \alpha\left(p_{c_k} p_{c_k}(\mathbf{x}) - S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right]\right)\right] - S_\alpha\left[\!\left[p_z \cdot p_z(\mathbf{x})\right]\!\right]\right)
\end{aligned}
$$

Further, it is obvious that

$$\frac{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k} = -\frac{\partial(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \mathbf{w}_k}$$

holds.

Appendix B2 - Derivatives of the cost function with respect to θ and $\mathbf{w}_k$ (second part)

Since we derived all necessary terms, the derivative $\frac{\partial E_{\text{cost}}}{\partial \theta}$ of the cost function with respect to θ can be written as:

$$
\begin{aligned}
\frac{\partial E_{\text{cost}}}{\partial \theta} = \frac{1}{\zeta} \sum_{(\mathbf{x},y)\in X} \Big(& C_e \cdot E(\mathbf{x},y,\mathbf{w}) \cdot \text{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot \Big[1 - \text{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta))\Big] \\
& - C_r \cdot R(\mathbf{x},\mathbf{w}) \cdot \text{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot \Big[1 - \text{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))\Big]\Big).
\end{aligned} \tag{8}
$$

For the derivative $\frac{\partial E_{\text{cost}}}{\partial \mathbf{w}_k}$ we have to consider two cases. In the case of $c_k \neq y$, we obtain the following formula:

$$\begin{aligned}
\frac{\partial E_{\text{cost}}}{\partial \mathbf{w}_k} = \sum_{\substack{(\mathbf{x},y)\in X \\ c_k \neq y}} \Bigg(& \Bigg[\frac{\partial H(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)} \cdot \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \cdot E(\mathbf{x},y,\mathbf{w}) \\
& \qquad + H(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot p_{c_k} \cdot \frac{\partial p_z(\mathbf{x})}{\partial \mathbf{w}_k} \Bigg] \cdot C_e \\
& + \Bigg[\frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))} \cdot (-1) \cdot \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \cdot R(\mathbf{x},\mathbf{w}) \\
& \qquad + H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot p_{c_k} \cdot \frac{\partial p_z(\mathbf{x})}{\partial \mathbf{w}_k} \Bigg] \cdot C_r \Bigg) \\
= \sum_{\substack{(\mathbf{x},y)\in X \\ c_k \neq y}} \Bigg(& \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \cdot \Bigg[\frac{\partial H(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)} \cdot E(\mathbf{x},y,\mathbf{w}) \cdot C_e \\
& \qquad - \frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))} \cdot R(\mathbf{x},\mathbf{w}) \cdot C_r \Bigg] \\
& + p_{c_k} \cdot \frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k} \cdot [C_e \cdot H(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) + C_r \cdot H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))] \Bigg)
\end{aligned} \tag{9}$$

In case of $c_k = y$ it follows that $\frac{\partial E(\mathbf{x},y,\mathbf{w})}{\partial \mathbf{w}_k} = 0$ in (7) and, hence,

$$\begin{aligned}
\frac{\partial E_{\text{cost}}}{\partial \mathbf{w}_k} = \sum_{\substack{(\mathbf{x},y)\in X \\ c_k = y}} \Bigg(& \Bigg[\frac{\partial H(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)} \cdot \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \cdot E(\mathbf{x},y,\mathbf{w}) + 0 \Bigg] \cdot C_e \\
& + \Bigg[\frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))} \cdot (-1) \cdot \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \cdot R(\mathbf{x},\mathbf{w}) \\
& \qquad + H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot p_{c_k} \cdot \frac{\partial p_z(\mathbf{x})}{\partial \mathbf{w}_k} \Bigg] \cdot C_r \Bigg) \\
= \sum_{\substack{(\mathbf{x},y)\in X \\ c_k = y}} \Bigg(& \frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \cdot \Bigg[\frac{\partial H(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)} \cdot E(\mathbf{x},y,\mathbf{w}) \cdot C_e \\
& \qquad - \frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))} \cdot R(\mathbf{x},\mathbf{w}) \cdot C_r \Bigg] \\
& + p_{c_k} \cdot \frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k} \cdot C_r \cdot H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \Bigg)
\end{aligned} \tag{10}$$

holds. Both derivatives can be combined to generate a batch update.

Summarizing, we can also write down the online-learning updates for a given

single data sample $(\mathbf{x}, y)$. In this case the prototype update rewrites as

$$\begin{aligned}\Delta \mathbf{w}_k \propto \Bigg(\frac{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)}{\partial \mathbf{w}_k} \cdot \Bigg[& \frac{\partial H(\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial \tilde{m}(\mathbf{x},\mathbf{w},\theta)} \cdot E(\mathbf{x},y,\mathbf{w}) \cdot C_e \\ & - \frac{\partial H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))}{\partial(-\tilde{m}(\mathbf{x},\mathbf{w},\theta))} \cdot R(\mathbf{x},\mathbf{w}) \cdot C_r \Bigg] \\ + p_{c_k} \cdot \frac{\partial p_{c_k}(\mathbf{x})}{\partial \mathbf{w}_k} \cdot & \left(\delta_y^{c_k} \cdot C_e \cdot H(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) + C_r \cdot H(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \right) \Bigg)\end{aligned} \tag{11}$$

as a single step update related to (9) and (10) whereas the threshold online update becomes

$$\begin{aligned}\Delta\theta \propto \frac{1}{\zeta} \Big(& C_e \cdot E(\mathbf{x},y,\mathbf{w}) \cdot \mathrm{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot \Big[1 - \mathrm{sig}_\zeta(\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \Big] \\ & - C_r \cdot R(\mathbf{x},\mathbf{w}) \cdot \mathrm{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \cdot \Big[1 - \mathrm{sig}_\zeta(-\tilde{m}(\mathbf{x},\mathbf{w},\theta)) \Big] \Big)\end{aligned} \tag{12}$$

according to (8). Thus all derivatives are available using the approximations (4) and (5) for the Heaviside and sigmoid function, respectively.

References

[1] M. E. Hellman. The nearest neighbor classification rule with a reject option. *IEEE Transactions on Systems Science and Cybernetics*, 6:179–185, 1970.

[2] R. Herbei and M. H. Wegkamp. Classification with reject option. *The Canadian Journal of Statistics*, 34(4):709–721, 2006.

[3] L. Fischer, B. Hammer, and H. Wersing. Rejection strategies for learning vector quantization. In M. Verleysen, editor, *Proc. of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, pages 41–46, Louvain-La-Neuve, Belgium, 2014. i6doc.com.

[4] T. C. W. Landgrebe, D. Tax, P. Paclìk, and R. P. W. Duin. The interaction between classification and reject performance for distance-based reject-option classifiers. *Pattern Recognition Letters*, 27:908–917, 2006.

[5] P. L. Bartlett and M. H. Wegkamp. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9:1823–1840, 2008.

[6] M. Yuan and M. H. Wegkamp. Classification methods with reject option based on convex risk minimization. *Journal of Machine Learning Research*, 11:111–130, 2010.

[7] A. Vailaya and A. K. Jain. Reject option for VQ-based Bayesian classification. In *Int. Conference on Pattern Recognition (ICPR)*, pages 2048–2051, 2000.

[8] K. Urahama and Y. Furukawa. Gradient descent learning of nearest neighbor classifiers with outlier rejection. *Pattern Recognition*, 28(5):761–768, 1995.

[9] C. K. Chow. An optimum character recognition system using decision functions. *IRE Transactions on Electronic Computers*, EC-6:247–254, 1957.

[10] C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions in Information Theory*, 16(1):41–46, 1970.

[11] T. Villmann, M. Kaden, D. Nebel, and M. Biehl. Learning vector quantization with adaptive cost-based outlier-rejection. In G. Azzopardi and N. Petkov,

editors, *Proceedings of 16th International Conference on Computer Analysis of Images and Pattern, CAIP 2015, Valetta - Malta*, volume Part II of *LNCS 9257*, pages 772 – 782, Berlin-Heidelberg, 2015. Springer.

[12] T. Villmann, M. Kaden, A. Bohnsack, S. Saralajew, and B. Hammer. Self-adjusting reject options in prototype based classification. In E. Merényi and M. Mendenhall and P. O'Driscoll, editors, *Advances in Self-Organizing Maps and Learning Vector Quantization: Proceedings of 11th International Workshop WSOM 2016*, Advances in Intelligent Systems and Computing, vol. 428, pages 269–279, Berlin-Heidelberg, 2016. Springer.

[13] Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, 1995. (Second Extended Edition 1997).

[14] S. Seo and K. Obermayer. Soft learning vector quantization. *Neural Computation*, 15:1589–1604, 2003.

[15] M. Kaden, W. Hermann, and T. Villmann. Attention based classification learning in GLVQ and asymmetric classification error assessment. In T. Villmann, F.-M. Schleif, M. Kaden, and M. Lange, editors, *Advances in Self-Organizing Maps and Learning Vector Quantization: Proceedings of 10th International Workshop WSOM 2014, Mittweida*, volume 295 of *Advances in Intelligent Systems and Computing*, pages 77–88, Berlin, 2014. Springer.

[16] L.v.d. Maaten and G. Hinten. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[17] T. Villman, M. Kaden, D. Nebel, and A. Bohnsack. Data similarities, dissimilarities and types of inner products - a mathematical characterization in the context of machine learning. *Machine Learning Reports*, 7(MLR-04-2015):18–28, 2015. http://www.techfak.uni-bielefeld.de/~fschleif/mlr/mlr_04_2015.pdf.

[18] M. Lange, D. Zühlke, O. Holz, and T. Villmann. Applications of l_p-norms and their smooth approximations for gradient based learning vector quantization.

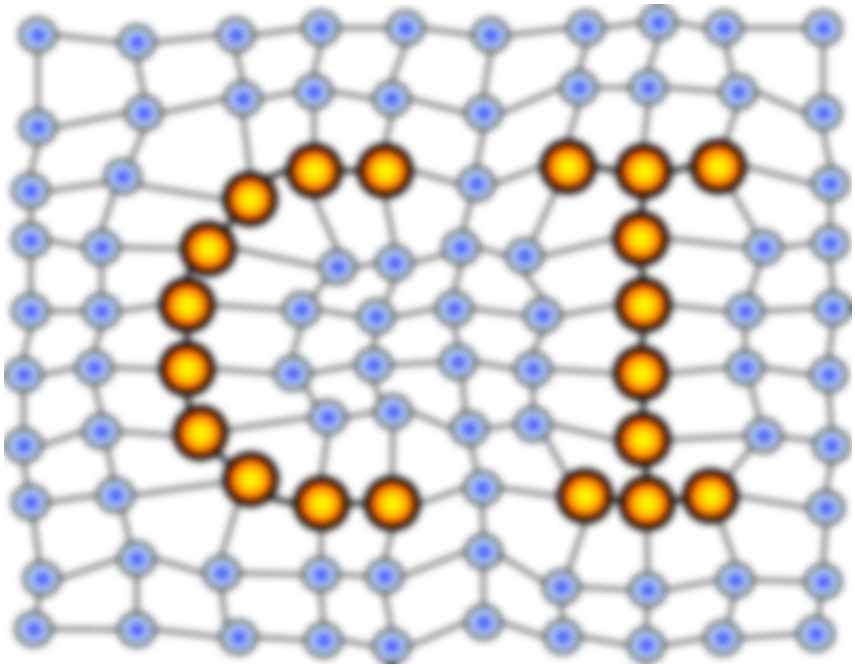
In M. Verleysen, editor, *Proc. of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, pages 271–276, Louvain-La-Neuve, Belgium, 2014. i6doc.com.

[19] M. Lange and T. Villmann. Derivatives of l_p-norms and their approximations. *Machine Learning Reports*, 7(MLR-04-2013):43–59, 2013. http://www.techfak.uni-bielefeld.de/~fschleif/mlr/mlr_04_2013.pdf.

[20] Thomas M. Martinetz, Stanislav G. Berkovich, and Klaus J. Schulten. 'Neural-gas' network for vector quantization and its application to time-series prediction. *IEEE Trans. on Neural Networks*, 4(4):558–569, 1993.

MACHINE LEARNING REPORTS

Report 01/2016



Impressum

Machine Learning Reports ISSN: 1865-3960

▽ Publisher/Editors

Prof. Dr. rer. nat. Thomas Villmann
University of Applied Sciences Mittweida
Technikumplatz 17, 09648 Mittweida, Germany
• http://www.mni.hs-mittweida.de/

Dr. rer. nat. Frank-Michael Schleif
University of Bielefeld
Universitätsstrasse 21-23, 33615 Bielefeld, Germany
• http://www.cit-ec.de/tcs/about

▽ Acknowledgments

We would like to thank the reviewers for their time and patience.